UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/672,517	09/27/2000	Richard K. Greicar	Gensp035	2127

22434 7590 09/22/2004

BEYER WEAVER & THOMAS LLP
P.O. BOX 778
BERKELEY, CA 94704-0778

EXAMINER	
BAKER, PAUL A	

ART UNIT	PAPER NUMBER
2188	

DATE MAILED: 09/22/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/672,517	GREICAR, RICHARD K.	
	Examiner	Art Unit	
	Paul A Baker	2188	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 August 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-51* is/are pending in the application.

4a) Of the above claim(s) *1-19* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *20-51* is/are rejected.

7)☒ Claim(s) *20,31,42 and 51* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *25 August 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 24 August 2004 has been entered.

Claim Objections

Claims 20 and 31 are objected to because of the following informalities: In the determining step "semifore" should be "semaphore".

Claim 42 is objected to because of the following informalities: claim ends in a double period.

Claim 51 is objected to because of the following informalities: claim ends with no period.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Claims 20-51 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Independent claims 20, 31, 42 and 47 recite "distributed processing by a processor" which is indefinite. Webopedia defines distributed processing as "any variety of computer systems that use more than one computer, or processor, to run an application." Therefore performing distributed processing by a processor is inconsistent with its meaning in the art.

For the purpose of examination of claims 20-41 "by a processor" in the preamble is not considered and all instances of "by the processor" will be interpreted as "by a processor". The reason the examiner has chosen an interpretation contrary to applicant's intended meaning (as stated in the remarks filed in after final amendment filed 15 June 2004, which has not been entered but is part of the record) is that the amended claims fail to properly restrict the claims in the manner applicant intended (See following preface to claims). Examiner must view all claims under the broadest possible interpretation. Since claims 20 and 31, as currently presented, more closely resemble examiner's stated interpretation and this is the broader of the two interpretations, this interpretation will be used as the basis for rejection.

For the purpose of examination of claims 42-51, applicant properly (as per applicant's intent) restricts the claims to a single processor system, therefore the limitation "distributed" is not considered.

Preface to Claim Rejections

A brief explanation of examiner's method of rejecting applicant's claimed invention is provided to clarify the record.

Claims 20 and 31 unsuccessfully attempt to restrict applicant's invention to a uniprocessor system, inclusion of the limitation "by the processor" does not preclude the existence of other processors within the system (limitation is not exclusionary). Therefore Scales remains anticipatory of applicant's claimed invention.

The examiner asserts that at the time of Scales' invention all major operating systems were multithreaded (Windows, MacOS, Linux, Unix, AIX, OS/2). Since all processes on a processor would be equally enabled to participate in Scales' invention, Scales anticipated his system being used in the situation where two processes on the same processor operate in the manner specified by Scales.

Claims 20-25 and 31-36 have been rejected twice, the first set uses Scales as basis for rejection, as applied in the final rejection mailed 3 June 2004, since examiner asserts that Scales still anticipates applicant's claimed invention. Assuming arguendo, claims 20-25,31-36, 42-51 are also rejected using Tanenbaum et al. "Operating Systems: Design and Implementation." Tanenbaum discloses applicant's claimed invention in a single processor system (Tanenbaum discloses that the methodology is also applicable for multiple processors, but embodies the concept within a single processor system).

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 20-26, 30-37, 41 are rejected under 35 U.S.C. 102(b) as being anticipated by Scales US Patent 5,761,729.

In regards to claim 20, Scales discloses a method of distributed processing, comprising:

providing a processor column 3 lines 44-45;

providing a memory having a plurality of memory segments capable of storing either program code or data column 3 lines 53-56;

providing a storage location for capable of storing semaphore values each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by a processor column 5 lines 56-62 in conjunction with column 8 line 49 through column 9 line 6;

providing a first program and a second program each operable to access the semaphore values in column 4 lines 28-40;

accessing a first semaphore value by said first program;

determining if the program code or data in the memory segment associated with the first semaphore value is available for use by a processor based upon the first semaphore value; and

using the first program to implement the code or data stored in the memory segment associated with the first semaphore value by a processor in column 4 lines 28-40.

In regards to claim 21, Scales discloses altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first semaphore value is available for having program code or data stored therein in column 3 lines 1-8.

In regards to claim 22, Scales discloses accessing the altered first semaphore value by the second program;

determining if the memory segment associated with the altered first semaphore value is available to have program code or data stored therein by the second program; and using the second program to store code or data in the memory segment associated with the semaphore value when the associated memory segment is available in column 3 lines 1-8 and column 4 lines 28-40.

In regards to claim 23, Scales discloses completing the storing of the program code or data in the memory segment associated with the semaphore value by the second program; and

altering the semaphore value by the second program to indicate that the program code or data in the memory segment associated with the first semaphore value is available for use by the first program in column 3 lines 1-8 and column 4 lines 28-.

In regards to claim 24, Scales discloses the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed by the processor in a distributed memory system disclosed by Scales in figure 2 element 216 is local to left most processor.

In regards to claim 25, Scales discloses the storage location is a register, or portions thereof, of the processor or a scalar accessible to the processor in column 8 lines 49 through column 9 line 5.

In regards to claim 26, Scales indirectly discloses the first program is a routine that is located in a reserved portion of the local memory such that the first program can not be written over with new code or data, and wherein the first program is operable to access the code or data stored in the local memory of the processor as well as implement that accessed code or data in column 3 lines 41-63 by stating that the process sets up a shared memory area to pass information from one process to another. It would be logical to conclude that Scales intents the first program is placed in

a non shared memory location, otherwise setting up a shared area would be unnecessary since the program would use its own variable space to convey information.

In regards to claim 30, Scales discloses the determining further comprises:

comparing the first semaphore value with a lookup list of pre-determined semaphore values in column 2 lines 66-67.

In regards to claim 31, Scales discloses a computer program product for distributed processing, comprising:

computer code for storing either program code or data in a memory having a plurality of memory segments column 3 lines 53-56;

computer code for providing a storage location capable of storing semaphore values each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by a processor column 5 lines 56-62 in conjunction with column 8 line 49 through column 9 line 6;

computer code for providing a first program and a second program each operable to access the semaphore values column 4 lines 28-40;

computer code for accessing a first semaphore value by said first program;

computer code for determining if the program code or data in the memory segment

associated with the first semaphore value is available for use by a processor based upon the first semaphore value;

computer code for using the first program to implement the code or data stored in the memory segment associated with the first semaphore value by a processor column 4 lines 28-40; and

computer readable medium for storing the computer code in column 2 line 66-column 3 line 8.

In regards to claim 32, Scales discloses computer code for altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first semaphore value is available for having program code or data stored therein in column 3 lines 1-8.

In regards to claim 33, Scales discloses computer code for accessing the altered first semaphore value by the second program;

computer code for determining if the memory segment associated with the altered first semaphore value is available to have program code or data stored therein by the second program; and

computer code for using the second program to store code or data in the memory segment associated with the semaphore value when the associated memory segment is available in column 3 lines 1-8 and column 4 lines 28-40.

In regards to claim 34, Scales discloses computer code for completing the storing of the program code or data in the memory segment associated with the semaphore value by the second program; and

computer code for altering the semaphore value by the second program to indicate that the program code or data in the memory segment associated with the first semaphore value is available for use by the first program in column 3 lines 1-8 and column 4 lines 28-40.

In regards to claim 35, Scales discloses the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed by the processor in figure 2 element 216 which is local to left most processor.

In regards to claim 36, Scales discloses a computer program product as recited in claim 31 wherein the storage location is a register, or portions thereof, of the processor or a scalar accessible to the processor in column 8 lines 49 through column 9 line 5.

In regards to claim 37, Scales discloses the first program is a routine that is located in a reserved portion of the local memory such that the first program can not be written over with previously presented code or data, and wherein the first program is operable to access the code or data stored in the local memory of the processor as well

as implement that accessed code or data in column 3 lines 41-63 by stating that the process sets up a shared memory area to pass information from one process to another. It would be logical to conclude that Scales intents the first program is placed in a non shared memory location, otherwise setting up a shared area would be unnecessary since the program would use its own variable space to convey information.

In regards to claim 41, Scales discloses the determining further comprises:

computer code for comparing the first semaphore value with a lookup list of pre-determined semaphore values in column 2 lines 66-67.

Claims 20-25,31-36, 42-51 are rejected under 35 U.S.C. 102(b) as being anticipated by Tanenbaum et al. "Operating Systems: Design and Implementation."

In regards to claim 20, Tanenbaum discloses a method of distributed processing, comprising:

providing a memory having a plurality of memory segments capable of storing either program code or data (code in figure 2-19, database which is contained within a portion (segment) of memory);

providing a storage location for capable of storing semaphore values each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by a processor (figure 2-19, line 3);

providing a first program and a second program each operable to access the semaphore values (figure 2-19, lines 6-20 and lines 23-31);

accessing a first semaphore value by said first program (figure 2-19, line 11);

determining if the program code or data in the memory segment associated with the first semaphore value is available for use by a processor based upon the first semaphore value (page 68 1st paragraph, last sentence and §2.2.5 2nd paragraph); and

using the first program to implement the code or data stored in the memory segment associated with the first semaphore value by a processor (figure 2-19, line 13).

In regards to claim 21, Tanenbaum discloses altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first semaphore value is available for having program code or data stored therein (figure 2-19, line 16).

In regards to claim 22, Tanenbaum discloses accessing the altered first semaphore value by the second program (figure 2-19, line 27);

determining if the memory segment associated with the altered first semaphore value is available to have program code or data stored therein by the second program (§2.2.5 2nd paragraph); and

using the second program to store code or data in the memory segment associated with the semaphore value when the associated memory segment is available (figure 2-19, line 28).

In regards to claim 23, Tanenbaum discloses completing the storing of the program code or data in the memory segment associated with the semaphore value by the second program (figure 2-19, line 28); and

altering the semaphore value by the second program to indicate that the program code or data in the memory segment associated with the first semaphore value is available for use by the first program (figure 2-19, line 29).

In regards to claim 24, Tanenbaum does not directly disclose the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed by the processor. However Tanenbaum embodies the invention in a single processor system, therefore the system memory is local to the processor. Additionally, Tanenbaum's disclosure is embodied within the Minix system, which employs virtual memory system, wherein memory is divided into a plurality of memory segments called pages (pages 319-320) these pages are addressed by the processor using their logical memory address.

In regards to claim 25, Tanenbaum discloses the storage location is a register, or portions thereof, of the processor or a scalar accessible to the processor (figure 2-19, line 3).

In regards to claim 31, Tanenbaum discloses a computer program product for distributed processing, comprising:

computer code for storing either program code or data in a memory having a plurality of memory segments (code in figure 2-19, database which is contained within a portion (segment) of memory);

computer code for providing a storage location capable of storing semaphore values each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by a processor (figure 2-19, line 3);

computer code for providing a first program and a second program each operable to access the semaphore values (figure 2-19, lines 6-20 and lines 23-31);

computer code for accessing a first semaphore value by said first program (figure 2-19, line 11);

computer code for determining if the program code or data in the memory segment associated with the first semaphore value is available for use by a processor based upon the first semaphore value (page 68 1st paragraph, last sentence and §2.2.5 2nd paragraph);

computer code for using the first program to implement the code or data stored in the memory segment associated with the first semaphore value by a processor (figure 2-19, line 13); and

computer readable medium for storing the computer code (memory in which the program is stored).

In regards to claim 32, Tanenbaum discloses computer code for altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first semaphore value is available for having program code or data stored therein (figure 2-19, line 16).

In regards to claim 33, Tanenbaum discloses computer code for accessing the altered first semaphore value by the second program (figure 2-19, line 27);

computer code for determining if the memory segment associated with the altered first semaphore value is available to have program code or data stored therein by the second program (§2.2.5 2nd paragraph); and

computer code for using the second program to store code or data in the memory segment associated with the semaphore value when the associated memory segment is available (figure 2-19, lines 27-28).

In regards to claim 34, Tanenbaum discloses computer code for completing the storing of the program code or data in the memory segment associated with the semaphore value by the second program (figure 2-19, line 28); and

computer code for altering the semaphore value by the second program to indicate that the program code or data in the memory segment associated with the first semaphore value is available for use by the first program (figure 2-19, line 29).

In regards to claim 35, Tanenbaum does not directly disclose the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed by the processor. However Tanenbaum embodies the invention in a single processor system, therefore the system memory is local to the processor. Additionally, Tanenbaum's disclosure is embodied within the Minix system, which employs virtual memory system, wherein memory is divided into a plurality of memory segments called pages (pages 319-320) these pages are addressed by the processor using their logical memory address.

In regards to claim 36, Tanenbaum discloses a computer program product as recited in claim 31 wherein the storage location is a register, or portions thereof, of the processor or a scalar accessible to the processor (figure 2-19, line 3).

In regards to claim 42, Tanenbaum discloses a single processor system that includes a memory coupled to the processor having a plurality of memory segments capable of storing either program code or data, a method of distributed processing by the processor (code in figure 2-19, database which is contained within a portion (segment) of memory), comprising:

storing semaphore values in the memory each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by the processor (figure 2-19, line 3);

accessing a first semaphore value by a first program (figure 2-19, line 11);

based upon the first semaphore value, determining if the program code or data in the memory segment associated with the first semaphore value is available for use by the processor (page 68 1st paragraph, last sentence and §2.2.5 2nd paragraph);

and implementing the code or data stored in the memory segment associated with the first semaphore value by the processor using the first program (figure 2-19, line 13).

In regards to claim 43, Tanenbaum discloses altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first semaphore value is available for having program code or data stored therein (figure 2-19, line 16).

In regards to claim 44, Tanenbaum discloses storing code or data in the memory segment associated with the first semaphore value with the altered semaphore value indicates that the memory segment is available by a second program (figure 2-19, lines 27-28).

In regards to claim 45, Tanenbaum discloses altering the semaphore value by the second program to indicate that the program code or data stored in the memory segment associated with the first semaphore value is available for use by the first program (figure 2-19, line 29).

In regards to claim 46, Tanenbaum does not directly disclose the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed. However Tanenbaum embodies the invention in a single processor system, therefore the system memory is local to the processor. Additionally, Tanenbaum's disclosure is embodied within the Minix system, which employs virtual memory system, wherein memory is divided into a plurality of memory segments called pages (pages 319-320) these pages are addressed by the processor using their logical memory address.

In regards to claim 47, Tanenbaum discloses a single processor system that includes a memory coupled to the processor having a plurality of memory segments

capable of storing either program code or data, computer program product for distributed processing by the processor (code in figure 2-19, database which is contained within a portion (segment) of memory), comprising:

computer code for storing semaphore values in the memory each of which are associated with one of said memory segments and operable to indicate whether said associated memory segment contains program code or data that is available for use by the processor (figure 2-19, line 3);

computer code for accessing a first semaphore value by a first program (figure 2-19, line 11);

computer code for determining if the program code or data in the memory segment associated with the first semaphore value is available for use by the processor based upon the first semaphore value (page 68 1st paragraph, last sentence and §2.2.5 2nd paragraph);

computer code for implementing the code or data stored in the memory segment associated with the first semaphore value by the processor using the first program (figure 2-19, line 13); and

computer readable medium for storing the computer code (memory in which the program is stored).

In regards to claim 48, Tanenbaum discloses altering the first semaphore value by the first program so as to indicate that the memory segment associated with the first

semaphore value is available for having program code or data stored therein (figure 2-19, line 16).

In regards to claim 49, Tanenbaum discloses computer code for storing code or data in the memory segment associated with the first semaphore value with the altered semaphore value indicates that the memory segment is available by a second program (figure 2-19, lines 27-28).

In regards to claim 50, Tanenbaum discloses computer code for altering the semaphore value by the second program to indicate that the program code or data stored in the memory segment associated with the first semaphore value is available for use by the first program (figure 2-19, line 29).

In regards to claim 51, Tanenbaum does not directly disclose the memory is a local memory of the processor and wherein the memory segments are logical memory segments implemented by the processor based upon a computer program to be executed. However Tanenbaum embodies the invention in a single processor system, therefore the system memory is local to the processor. Additionally, Tanenbaum's disclosure is embodied within the Minix system, which employs virtual memory system, wherein memory is divided into a plurality of memory segments called pages (pages 319-320) these pages are addressed by the processor using their logical memory address.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 26-29 are rejected under 35 U.S.C. 103(a) as being unpatentable over Scales US Patent 5,761,729.

In regards to claim 27, Scales does not disclose the program that is operable to implement a portion of a Fast Fourier Transform (FFT) program is stored in the local memory, the first program is operable to access the local memory and begin implementing the FFT code, however Scales discloses a general purpose method of passing data from one process to another. It is well known in the art of Digital Signal Processing the use of a Consumer-Producer model for signal processing wherein one process loads and stores FFT data to be operated in a shared memory space and for another process to process the data located in the shared memory space, Therefore the examiner takes official notice.

In regards to claim 28, Scales does not disclose the second program is operable to load new blocks of code or data that are used by the first program, however Scales discloses a general purpose method of passing data from one process to another. It is well known in the art of Digital Signal Processing the use of a Consumer-Producer

model for signal processing wherein one process loads and stores FFT data to be operated in a shared memory space and for another process to process the data located in the shared memory space, Therefore the examiner takes official notice.

In regards to claim 29, Scales discloses the second program loads code or data from an external memory to the local memory based upon the first semaphore value in column 3 lines 1-8.

In regards to claim 38, Scales does not disclose the program that is operable to implement a portion of a Fast Fourier Transform (FFT) program is stored in the local memory, the first program is operable to access the local memory and begin implementing the FFT code, however Scales discloses a general purpose method of passing data from one process to another. It is well known in the art of Digital Signal Processing the use of a Consumer-Producer model for signal processing wherein one process loads and stores FFT data to be operated in a shared memory space and for another process to process the data located in the shared memory space, Therefore the examiner takes official notice.

In regards to claim 39, Scales does not disclose the second program is operable to load previously presented blocks of code or data that are used by the first program, however Scales discloses a general purpose method of passing data from one process to another. It is well known in the art of Digital Signal Processing the use of a

Consumer-Producer model for signal processing wherein one process loads and stores FFT data to be operated in a shared memory space and for another process to process the data located in the shared memory space, Therefore the examiner takes official notice.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Paul A Baker whose telephone number is (703)305-3304. The examiner can normally be reached on M-F 10am-6:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Mano Padmanabhan can be reached on (703)306-2903. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

PB

Mano Padmanabhan 9/17/04

MANO PADMANABHAN
SUPERVISORY PATENT EXAMINER